doi: 10.7690/bgzdh.2019.02.011

高冲突性证据的软件可信性评估方法

陈倩倩，许丽星，龚　彬
（中国工程物理研究院电子工程研究所，四川 绵阳 621999）

摘要：为解决传统基于 DS 理论的融合算法在处理高冲突证据时，存在将所有证据都视为高冲突证据统一进行修正的问题，提出一种处理证据高冲突性的改进算法。判断证据的冲突程度，只对冲突性较大的证据进行修正，在保证原始证据真实性的基础上对高冲突证据源进行修正，用 DS 合成法则进行融合，并通过实验验证。实验结果表明：该算法具有良好的收敛性和可行性，与其他算法相比，可在一定程度上提高合成过程中证据的可靠性。

关键词：可信软件；评估方法；高冲突证据；证据理论
中图分类号：TP311　　**文献标志码：**A

Software Trustworthy Evaluation for High-conflict Evidence Reasoning

Chen Qianqian, Xu Lixing, Gong Bin
(*Institute of Electronic Engineering*, *China Academy of Engineering Physics*, *Mianyang* 621999, *China*)

Abstract: In order to solve the problem that the traditional fusion algorithm based on DS theory treats all the evidences as high conflict evidences in the process of handling them, this paper proposes an improved algorithm to deal with the high conflict of evidences. We judge the conflicting degrees of evidences firstly, then correcting the high conflict of evidences to reduce them based on the authenticity of original evidences and using DS synthesis rule to fuse the evidences, and then verifying the rationality through the experiment. The results of experiment show that the algorithm we proposed has good convergence and feasibility. Compared with other algorithms, the proposed algorithm can improve the reliability of evidence to some extent in the synthesis process.

Keywords: trustworthy software; evaluation method; high-conflict evidence; evidential theory

0　引言

当今社会，计算机软件已被广泛应用于各个领域中，若这些软件发生失效，会给社会和经济带来极大的损失，严重影响人类的正常生活；因此，人们迫切地想通过软件评估来了解软件是否可信，避免低可信性软件应用于安全关键领域而出现的严重后果。如何评估软件可信是一个复杂的科学问题，多属性、主观性和演化性等一系列特点，使得传统基于统计学的时域模型评估方法在工程应用中遇到了瓶颈；因此，举证法被提了出来。举证法以实际项目的开发数据为评估基础，较为全面地考虑了软件执行过程中影响其可信性的各种因素，通过收集软件生命周期中能够证明软件可信的各种证据，通过特定算法进行融合，从而判断其可信程度。该方法能够达到量化评估软件可信性的目的，在一定程度上避免了可信评估中由于评估人员主观定性判断所带来的评估方法不客观、片面性强和评估结果不准确等问题。

用举证法对软件可信性进行评估，收集软件生命周期中各种可信证据是进行软件评估的基础和前提。只有保证证据的准确性与可靠性，才能保证最后评估结果的准确性。在此基础上，还要对收集到的证据进行融合。所谓的证据融合，是指在对软件可信性进行评估时，如何将收集到的多个证明软件可信程度的证据进行融合，使其结果可以直观反映出软件的可信级别。

目前已有学者对软件可信性评估算法进行了研究。杜晶等[1]提出了一种基于证据的可信软件过程评估方法，利用可信基线数据判断软件过程的可信性，以实际过程数据为证据，在一定程度上保证了检测过程的客观性与全面性，但对可信证据的赋值只有 0 与 1 而没有中间值，与实际测量数据不太相符；杨善林等[2]为适应复杂环境下软件可信性评估的需求提出了一种基于效用和证据理论的可信软件评估方法，以实测证据的置信度为依据，用证据理论进行融合，但没有考虑到证据理论的合成悖论等问题；丁帅等[3]提出了一种基于需求动态演化的软件可信性评估方法，能够根据需求来配置软件评估

收稿日期：2018-11-15；**修回日期：**2018-12-22
基金项目：中物院十三五重大预研项目（TA060607）
作者简介：陈倩倩（1994—），女，河南人，硕士，从事计算机通信与控制研究。

准则体系，但需提前知道软件运行状态；文献[4]提出了基于DS证据理论的软件可信性分类研究方法，在用可信证据证明软件可信程度的基础上采用Shannon 熵来表示评估结果的不确定性，却没有说明如何降低这种不确定性；文献[5]从不同角度提出了一种基于属性的软件可信级别分配方法，反面给出了若软件要达到一定的可信等级，则属性层面上可信性要达到多少，却无法对已知软件的可信性进行评估。

总的来说，因证据理论具有简单易懂、便于计算的特点，能够统一识别框架上的多个不同可信证据，考虑用证据理论对可信证据进行融合以达到可信评估是合理且可行的，但在该过程中还存在证据的高冲突性问题。针对这一问题，普遍的做法是引入一个折扣系数对高冲突证据进行修正，虽然在一定程度上减轻了证据间的冲突性，却在细节处理上有一些不合理之处。首先是在对证据进行修正之前并没有判断哪些证据是高冲突证据，并针对高冲突性证据进行修正，而是对所有证据统一进行修正。其次是在修正过程中对同一证据识别框架内的全体度量元进行了同一程度的修正，这样相同程度的修正并不合理，因为识别框架内的每个度量元其冲突程度不一定完全相同(即证据收集过程中由于主客观因素导致证据的准确程度并不相同)；因此，笔者提出一种考虑证据高冲突性的可信证据合成方法，使得评估结果更加真实可信。

1 考虑证据高冲突性的软件可信评估算法

考虑证据高冲突性的软件可信评估算法，笔者通过改进算法对证据源进行修正，确保修正后证据源的准确性与可信性，再用证据理论合成规则对修正后的证据进行融合，进而使软件评估结果更接近真实的可信程度。

1.1 相关定义及概念

概念 1：设 θ 表示 H 所有可能取值的一个论域集合，且所有在 θ 内的元素间互不相容，则称 θ 是 H 的识别框架。

定义 1 设 θ 为一识别框架，则函数 m：$2^H \to [0,1]$ 在满足下列条件时，

1) $m(\phi)=0$，

2) $\sum_{A\in H} m(A)=1$，

称 $m(A)$ 为 A 的基本概率赋值，$m(A)$ 表示对 A 的直接支持。

概念 2：若识别框架 θ 的一个子集为 A，且 $m(A)>0$，则称 A 为信任函数 BEL 的焦元(Focal Element)，所有焦元的并称为核(Core)。

定义 2 设 BEL_1 和 BEL_2 是同一识别框架 θ 上的 2 个信任函数，m_1 和 m_2 分别为对应的基本概率赋值，焦元分别为 $A_1,A_2,\cdots,A_i$ 和 $B_1,B_2,\cdots,B_j$，又设

$$K_1=\sum_{\substack{i,j\\ A_i\cap B_j=\phi}} m_1(A_i)m_2(B_j)<1。\tag{1}$$

则

$$m_{1,2}(C)=\begin{cases}\dfrac{\sum\limits_{\substack{i,j\\ A_j\cap B_j=C}} m_1(A_i)m_2(B_j)}{1-K_1} & \forall C\subset U, C\neq\phi\\ 0 & C=\phi\end{cases}。\tag{2}$$

式中：若 $K_1\neq 1$，则 m 确定一个基本概率赋值；若 $K_1=1$，则认为 m_1 和 m_1 矛盾，不能组合。

定义 3 对于一个可信属性下的一组可信证据 $E_1,E_2,\cdots,E_i$，将该组证据焦元 H_j 的置信度 $\beta_{1j},\beta_{2j},\cdots,\beta_{ij}$ 作为每组证据的一组焦元度量值，计算每组焦元度量值的方差 var。若 var＜0.02，认为该组焦元度量值的离散程度较小，一致程度较高，定义该组数据所属焦元 H_j 为可信焦元；若 var＞0.02，则为不可信焦元。

定义 4[6] 设识别框架 $\theta=\{H_1,H_2,\cdots,H_n\}$，$\forall H_k$，2 条证据的基本概率分配函数分别为 $m_i(H_k)$、$m_j(H_k)$，则这 2 条证据关于 H_k 的相容系数为：

$$R_{i,j}(H_k)=\frac{2m_i(H_k)m_j(H_k)}{m_i(H_k)^2+m_j(H_k)^2}。\tag{3}$$

相容系数反映了证据两两间的相互支持程度，当 $m_i(H_k)$ 和 $m_j(H_k)$ 其中一个为 0 时，说明该证据完全否认 H_k，而另一个证据在一定程度上支持 H_k，此时两证据间高度冲突，相容系数为 0；当 $m_i(H_k)$ 和 $m_j(H_k)$ 相等时，说明两证据所持观点相同，具有高度一致性，此时相容系数为 1。相容系数较好地反映了证据间的相容性，用相容系数作为证据源的修正因子，对证据源的可信度进行重新分配，在一定程度上能够减轻证据源之间的冲突程度。

1.2 可信评估模型

软件可信性是一个综合指标，对该指标进行层层划分，即软件可信性—可信属性—可信子属性—可信证据，获取底层可信证据度量值，再通过证据理论融合算法逐步向上融合，得到最顶层的软件可

信评估结果。基于分层的可信评估模型如图 1 所示。

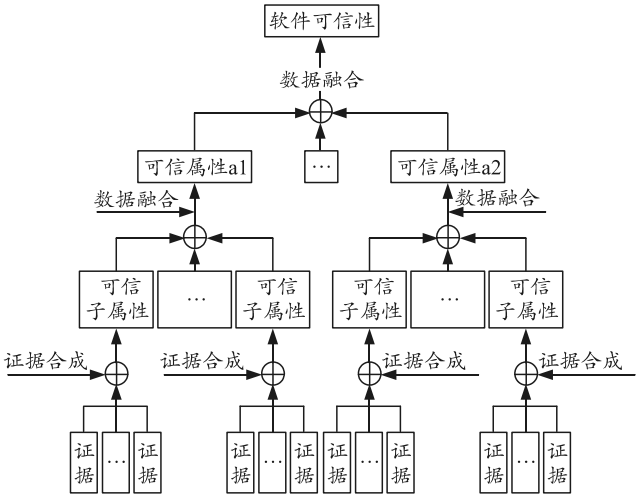

图 1　基于分层的可信评估模型

在软件可信评估模型中，可信证据位于评估模型的最底层，也是评估软件可信性的基础。在获取可信证据的过程中，采集到的证据种类多种多样，需要通过不同评价准则来获取相应的度量值。为了使获取的度量值具有相同量纲，方便后续数据进行融合，采用基于效用的信息转换技术[7]对可信证据度量值进行转换，使其具有同一量纲。

1.3　基于效用的可信度量值转换方法

效用值以评估专家对现状的精神感受值为基础，具有较强的客观性。在软件可信性评估过程中，若将采集到的度量指标值记为 H_n，则 H_n 对于评估专家的效用值可记为 $u(H_n)$。

对于定性指标度量值，通过加权效用函数可以计算出专家给出的评价的效用值[8]，λ_n 表示专家对该评估等级 H_n 的置信度。

$$u=\sum \lambda_n u(H_n)。\tag{4}$$

对于定量指标度量值，当获取定量指标 e 的分段效用函数后，原始评估数据 V_e 的效用 $u(V_e)$ 可由下式计算：

$$u(V_e)=u(H_{n+1})-\frac{V_{H_{n+1}}-V_e}{V_{H_{n+1}}-V_{H_n}}(u(H_{n+1})-u(H_n))。\tag{5}$$

由上式可得出原度量指标的效用值，再通过公式

$$\beta_n=\frac{H_{n+1}-u(e)}{H_{n+1}-H_n},\ \beta_{n+1}=1-\beta_n,\ H_n\leqslant u(e)\leqslant H_{n+1}$$

建立可信度量指标在可信评估效用等级 H_n 上置信度（其中 $u(H)=\{u(H_n),n=1,2,\cdots,5\}$ 由领域专家给出）。利用效用进行信息转换后，不仅可以将定性可信度量指标上的评价信息进行合理量化，而且能保证转换后的信息具有统一的量纲[9]。

1.4　可信评估算法

以 1.2 节中的可信评估模型为基础，对收集到的可信证据进行效用转换，得到可直接用于计算的可信度量值。

在对证据进行修正时，要首先判断证据的冲突程度，根据冲突程度的大小进行不同程度的修正。考虑到证据的期望值代表了证据度量值整体的平均情况，若证据本身与期望值相差越大，说明该证据与其他证据间的冲突性越大[10]。利用相容系数公式计算每个证据与期望之间的相容程度作为修正因子，相容系数越大，对应的修正因子就越大，证明该证据与均值的一致性越好，可信程度越高。

将收集到的各类可信证据根据可信属性进行分类。一般来说，每个可信属性下的一组证据之间应该相互支持，即该组证据在每个度量元下的一组度量值彼此间离散程度很小，数值很集中。因此，在对可信证据进行修正前，先对每组证据的离散程度进行判定。考虑到方差可以有效衡量一组数据的离散程度，方差越大，离散程度越大，方差越小，则数值越集中，用方差来对每组度量值的离散程度进行判断：若方差大于 0.02（经实际计算，当一组数据的方差小于 0.02 时，其离散程度较小，一致程度较高），视为该组度量值的离散程度大，可信程度低，需要进行修正；若方差小于 0.02，可认为其可信程度高，无需修正。

具体算法步骤如下：

1) 建立统一的软件可信评估识别框架 $\Omega=\{H_1,H_2,H_3,H_4,H_5\}$，令各评价等级的效用函数 $u(H)=\{u(H_n),n=1,2,\cdots,5\}$。

2) 利用方差计算公式计算每个可信属性下每组度量元的方差，方差计算如式(6)，其中：$\bar{\beta}$ 为期望，$\bar{\beta}=\dfrac{\beta_{i1}+\beta_{i2}+\ldots+\beta_{ij}}{j}$，若方差大于 0.02，按步骤 3)进行修正；否，则无需修正。

$$\mathrm{var}=\frac{(\beta_{i1}-\bar{\beta})^2+(\beta_{i2}-\bar{\beta})^2+\ldots+(\beta_{ij}-\bar{\beta})^2}{j}。\tag{6}$$

3) 计算每组需要修正的焦元度量值的均值，利用式(3)计算均值与每个焦元度量值的相容系数作为修正因子，将修正因子与相对应的焦元度量值相

乘，得到修正后的可信焦元度量值。为方便区分，笔者称修正后的度量值为可信度量值，修正后的可信度量值之间冲突性会有所减轻，离散程度更加集中，从而更加可信。

4) 利用式(7)计算 mass 函数，其中 ω 为每个证据所占权重，$\omega=1/i$，i 为每组可信证据的个数(因很难说明哪个证据更为重要，因此将所有证据权重视为相等)；

$$m=\omega*\beta_{ij},\quad m_{\Omega}=\bar{m}+\tilde{m}。\tag{7}$$

其中：m_{Ω} 为不确定性，由 $\bar{m}$ 和 $\tilde{m}$ 组成；$\bar{m}=1-\omega$ 表示由权重引起的不确定性；$\tilde{m}=\omega*(1-\sum_{i}\beta_{i,j})$ 表示由原始信息不完整引起的不确定性。

5) 利用 DS 合成规则(式(1)与式(2))对每个子属性下的一组可信证据进行融合，得到每个可信子属性的 mass 函数。

6) 再次通过 DS 合成规则层层向上进行融合，从而得到软件可信层面的 mass 函数，利用式(8)对评估结果进行实际效用计算(其中 m_{Ω} 为不确定程度，$|m|$为非 0 焦元个数)，得到最终的可信定量评估结果。

$$u(m)=\sum_{i}m_iu(H_i)+m_{\Omega}u(H_{\Omega})=\sum_{i}m_iu(H_i)+\sum_{m_i\neq0}\frac{m_{\Omega}u(H_i)}{|m|}。\tag{8}$$

2 实验案例分析

以医疗信息系统(hospital information system，HIS)软件为实验对象[10]，通过获取软件开发过程中的可信证据以及相应的可信度量值，用所提算法进行可信评估，证明该算法的合理性与有效性。

设识别框架 $\Omega=\{H_1,H_2,H_3,H_4,H_5\}$，其中 H_1 表示软件可信级别最低，从 H_1 到 H_5 逐步升高，H_5 表示软件可信程度最高。决策专家给出的效用函数为 $u(H_n)=\{0,0.25,0.5,0.75,1\}$，经效用转换后的可信证据指标度量值如表 1 所示。

表 1 HIS 软件可信评估原始数据及修正后的可信证据源

可信属性	度量值	原始可信证据(H_n,β_n)	修正后的可信证据(H_n,β_n)
可靠性(e_1)	成熟度(e_{11})	(H_4,0.81), (H_5,0.19), (Ω,0)	(H_4,0.782 9), (H_5,0.19), (Ω,0.271)
	容错设计(e_{12})	(H_3,0.2), (H_4,0.64), (Ω,0.16)	(H_3,0.197 7), (H_4,0.639 4), (Ω,0.162 9)
	运行效率(e_{13})	(H_3,0.5), (H_4,0.42), (Ω,0.08)	(H_3,0.383 2), (H_4,0.389 3), (Ω,0.227 5)
防危性(e_2)	软件生存率(e_{21})	(H_3,0.22), (H_4,0.78), (Ω,0)	(H_3,0.217 6), (H_4,0.774 2), (Ω,0.008 2)
	风险警告(e_{22})	(H_3,0.64), (H_5,0.36), (Ω,0)	(H_4,0.638 2), (H_5,0.216), (Ω,0.145 8)
	应急处理能力(e_{23})	(H_3,0.35), (H_4,0.65), (Ω,0)	(H_3,0.293 5), (H_4,0.648 8), (Ω,0.057 7)
实时性(e_3)	性能(e_{31})	(H_4,0.47), (H_5,0.53), (Ω,0)	(H_4,0.47), (H_5,0.53), (Ω,0)
	单位任务时间(e_{32})	(H_4,0.44), (H_5,0.29), (Ω,0.27)	(H_4,0.44), (H_5,0.29), (Ω,0.27)
可维护性(e_4)	设计规范化(e_{41})	(H_4,0.18), (H_5,0.82), (Ω,0)	(H_4,0.115 9), (H_5,0.491 2), (Ω,0.392 9)
	可理解性(e_{42})	(H_3,0.44), (H_4,0.56), (Ω,0)	(H_3,0.364 2), (H_4,0.555 5), (Ω,0.080 3)
	HIS 可替代性(e_{43})	(H_3,0.26), (H_4,0.74), (Ω,0)	(H_3,0.258 5), (H_4,0.683), (Ω,0.058 5)
可用性(e_5)	自恢复时间(e_{51})	(H_3,0.67), (H_4,0.16), (Ω,0.17)	(H_3,0.67), (H_4,0.16), (Ω,0.17)
	应急计划集成性(e_{52})	(H_3,0.62), (H_4,0.38), (Ω,0)	(H_3,0.62), (H_4,0.38), (Ω,0)
	任务执行准确率(e_{53})	(H_3,0.67), (H_4,0.33), (Ω,0)	(H_3,0.67), (H_4,0.33), (Ω,0)
安全性(e_6)	外部接入可控性(e_{61})	(H_4,0.41), (H_5,0.47), (Ω,0.12)	(H_4,0.41), (H_5,0.403 5), (Ω,0.186 5)
	隐私保护(e_{62})	(H_4,0.67), (H_5,0.33), (Ω,0)	(H_4,0.67), (H_5,0.322 6), (Ω,0.007 4)
	软件集成性(e_{63})	(H_3,0.65), (H_4,0.35), (Ω,0)	(H_3,0.39), (H_4,0.35), (Ω,0.26)

计算证据 e_1 中属于焦元 H_3 的一组数据(0,0.20,0.50)的方差，结果为 0.042 2＞0.02，需要进行修正。求出该组数据的均值等于 0.233 3，并计算每个数据与均值的相容系数：

$$R_{e_{11},H_3}=\frac{2\times0^2\times0.233\,3^2}{0^2+0.233\,3^2}=0;$$

$$R_{e_{12},H_3}=\frac{2\times0.2^2\times0.233\,3^2}{0.2^2+0.233\,3^2}=0.988\,3;$$

$$R_{e_{13},H_3}=\frac{2\times0.5^2\times0.233\,3^2}{0.5^2\times0.233\,3^2}=0.766\,4。$$

将原始数据与相容系数之积作为修正后的可信度量值，经计算分别为 0, 0.197 7, 0.383 2，与此类似，求出修正后的可信证据源如表 1 所示。

以修正后的可信证据源为基础，通过式(7)计算基本概率分配函数 m，并通过式(1)和式(2)对 6 组可信证据源进行合成，得到在可信属性上的基本概率

分配函数 m_i，如表 2 所示。

表 2　可信属性上证据融合后的基本概率分配函数

可信属性	Mass 函数(H_n,m_i)
可靠性	(H_3,0.112 5), (H_4,0.447), (H_5,0.037 7), (Ω,0.402 8)
防危性	(H_3,0.097 4), (H_4,0.499 6), (H_5,0.036 9), (Ω,0.366 1)
实时性	(H_3,0), (H_4,0.342 6), (H_5,0.307 6), (Ω,0.349 8)
可维护性	(H_3,0.140 5), (H_4,0.334 9), (H_5,0.089 3), (Ω,0.435 3)
可用性	(H_3,0.464), (H_4,0.175 9), (H_5,0), (Ω,0.360 1)
安全性	(H_3,0.072 6), (H_4,0.348 9), (H_5,0.157 8), (Ω,0.420 7)

以此为基础，再用 DS 合成规则对可信属性度量值进行融合，得到 HIS 软件综合可信评估结果为：$m=\{(H_3,0.086\,2),(H_4,0.845\,7),(H_5,0.05),(\Omega,0.018\,1)\}$。从评估结果中可以看出：文中算法的收敛性比较好，且 HIS 软件的可信级别属于 H_4 的可能性最高，再利用式(8)对评估结果进行实际效用计算，得到最终的可信综合评价结果为 $u(m)=0.741$，从而可知 HIS 软件的可信程度比较高，这与其自身的可信性较为相符，表明算法是可行的。

若不考虑证据高冲突性对评估结果的影响而直接进行融合，得到 HIS 软件的综合可信评估结果为：$m=\{(H_3,0.239\,1),(H_4,0.579\,8),(H_5,0.163\,9),(\Omega,0.017\,2)\}$，可信综合评价结果为 $u(m)=0.731\,2$，经对比可发现该方法的收敛性不如文中所提算法。

3　结束语

笔者提出了一种对高冲突性证据进行修正的改进算法。与其他软件可信评估算法相比，文中算法在一定程度上提高了合成过程中证据的可靠性，从而使评估结果更为客观准确。通过实际案例进行验证，并将实验结果与其他算法结果进行比较，证明该方法具有良好的收敛性与可行性。

在后期的研究中，应重点关注可信属性间的相关性对软件可信评估结果的影响，通过弱化属性间的相关性，以增强软件可信评估结果的准确性。

参考文献：

[1] 杜晶, 杨叶, 王青, 等. 基于证据的可信软件过程评估方法[J]. 计算机科学与探索, 2011, 5(6): 501−512.

[2] 杨善林, 丁帅, 褚伟. 一种基于效用和证据理论的可信软件评估方法[J]. 计算机研究与发展, 2009, 46(7): 1152−1159.

[3] 丁帅, 鲁付俊, 杨善林, 等. 一种需求驱动的软件可信性评估及演化模型[J]. 计算机研究与发展, 2011, 48(4): 647−655.

[4] WANG B, ZHOU X, YANG G, et al. DS Theory-Based Software Trustworthiness Classification Assessment[C]// Symposia and Workshops on Ubiquitous, Autonomic and Trusted Computing. IEEE Computer Society, 2010: 434−438.

[5] MA Y J, CHEN Y X. An Attributes-based Allocation Approach of Software Trustworthy Degrees[C]. 2015 IEEEE International Conference on Software Quality, Reliability and Security-Comoanion, 2015.

[6] 汪永伟, 刘育楠, 杨英杰, 等. D−S 证据理论中冲突性处理新方法[J]. 计算机工程与设计, 2013, 34(12): 4316−4320.

[7] YANG J B. Rule and utility based evidential reasoning approach for multi-attribute decision analysis under uncertainties[J]. European Journal of Operational Research, 2001, 131(1): 31−61.

[8] 孙媛, 赵建军, 周源. 基于软计算技术的军用软件可靠性预测模型研究[J]. 兵工自动化, 2017, 36(2): 56−60.

[9] 张卫祥, 刘文红, 吴欣. 软件可信性定量评估[M]. 北京: 清华大学出版社, 2015: 140−141.

[10] 朱友清, 周石琳, 邹焕新. 基于相容系数的冲突证据合成方法及评价准则[J]. 系统工程与电子技术, 2014, 36(6): 1118−1123.

[11] 丁帅. 软件可信性评估模型及其优化方法研究[D]. 合肥: 合肥工业大学, 2011: 33−34.